CLAIMS

What is claimed is:

1. A method of facilitating access to data, the method comprising:

providing each of a plurality of heterogeneous data sources with an associated software wrapper that provides an object representation of data in the data source;

providing outputs of one or more software wrappers to a first software accumulator that aggregates data from data sources to generate a first aggregate data representation; and

using at least a second software accumulator to generate a second aggregate data representation different from the first aggregate data representation based at least in part on the first aggregate data representation from the first software accumulator.

2. The method of claim 1 wherein at least one of the software wrappers hides one or more details of the data source.

3. The method of claim 2 wherein the one or more details hidden by the software wrapper comprise one or more of a data format of the data source and a location of the data source.

4. The method of claim 1 wherein the second aggregate data representation is generated using the first aggregate data representation from the first software accumulator and data from one or more software wrappers.

5. The method of claim 1 wherein the at least one software wrapper used to generate the second aggregate data representation also is used to generate first aggregate data representation.

6. The method of claim 1 wherein the at least at least one software wrapper used to generate the second aggregate data representation is different from the one or more software wrappers used to generate first aggregate data representation.

7. The method of claim 1 wherein the second aggregate data representation is generated using the first aggregate data representation from the first software accumulator and data from at least a third software accumulator.

8. The method of claim 1 further comprising interconnecting any arbitrary number of software accumulators to generate a corresponding number of aggregate data representations.

9. The method of claim 1 further comprising using aggregate data representations as building blocks to generate additional aggregate data representations as desired.

10. The method of claim 1 further comprising generating a universal data representation by normalizing the first or the second aggregate data representations.

11. The method of claim 1 further comprising caching information from one or more data sources.

12. The method of claim 11 wherein the information caching occurs at a software wrapper level or a software accumulator level or both.

13. A method of managing access to a data source, the method comprising:

encapsulating a data source in a software wrapper configured to accommodate one or more parameters of the data source and to provide an object representation of data in the data source;

detecting that one or more parameters of the data source have changed; and

automatically downloading from a remote source a replacement software wrapper configured to accommodate the changed one or more parameters of the data source.

14. The method of claim 13 further comprising installing the replacement software wrapper while the original software wrapper is executing.

15. The method of claim 13 wherein the one or more parameters of the data source relate to one or more of a format or a location of data in data source.

16. The method of claim 13 wherein the remote source comprises a self-healing manager component executing on a remote platform.

17. The method of claim 16 wherein the self-healing manager performs operations comprising:

determining whether a replacement software wrapper exists; and

if so, providing the replacement software wrapper to a requesting entity; and

if not, notifying a support site that a replacement software wrapper has been requested.

18. The method of claim 13 wherein detecting that one or more parameters of the data source have changed comprises identifying a change in the data that the software wrapper is unable to accommodate.

19. The method of claim 13 wherein automatically downloading a replacement software wrapper from a remote source comprises sending an error manager to a remote self-healing manager component.

20. The method of claim 13 further comprising, upon detecting that one or more parameters of the data source have changed, ceasing to provide data from the software wrapper.

21. The method of claim 20 further comprising:

installing the automatically downloaded software wrapper; and

resuming to provide data from the software wrapper without having to restart an application associated with the software data wrapper.

22. The method of claim 13 wherein automatically downloading a replacement software wrapper from a remote source comprises periodically polling a remote process until a replacement software wrapper is available.

23. A method of managing access to a data source, the method comprising:

encapsulating each of a plurality of data sources in an associated software wrapper configured to provide an object representation of data from the data source;

providing outputs of the software wrappers to a software accumulator that aggregates data to generate an aggregate data representation;

detecting that one or more data parameters have changed; and

automatically downloading from a remote source a replacement software accumulator configured to accommodate the changed one or more data parameters.

24. The method of claim 23 further comprising automatically downloading from a remote source a replacement software wrapper configured to accommodate the changed one or more data parameters.

25. The method of claim 23 further comprising installing the replacement software accumulator while the original software accumulator is executing.

26. The method of claim 23 wherein the one or more data parameters relate to one or more of a format or a location of data in data source.

27. The method of claim 23 wherein the remote source comprises a self-healing manager component executing on a remote platform.

28. The method of claim 23 wherein the self-healing manager performs operations comprising:

determining whether a replacement software accumulator exists; and

if so, providing the replacement software accumulator to a requesting entity; and

if not, notifying a support site that a replacement software accumulator has been requested.

29. The method of claim 23 further comprising, upon detecting that one or more data parameters have changed, ceasing to provide data from the software accumulator.

30. The method of claim 29 further comprising:

installing the automatically downloaded software accumulator; and

resuming to provide data from the software accumulator.

31. The method of claim 23 wherein automatically downloading a replacement software accumulator from a remote source comprises periodically polling a remote process until a replacement software accumulator is available.

32. A distributed data processing system comprising:

an interface configured to receive a data processing request from a requesting entity;

a processing server configured to provide access to one or more local data processing applications;

one or more shadow processing servers, each shadow processing server configured to provide access to one or more remote data processing applications; and

an application server, in communication with the processing server and the shadow processing server, and configured to fulfill the received data processing request by selectively accessing local and remote data processing applications in a manner that is transparent to the requesting entity.

33. The system of claim 32 wherein the interface configured to receive a data processing request from a requesting entity comprises a web server.

34. The system of claim 32 wherein each shadow processing server has a communications link for communicating with an interface at a remote data processing system.

35. The system of claim 34 wherein the shadow processing server communicates with a servlet executing in a web server at the remote data processing system.

36. The system of claim 32 wherein each shadow processing server has an associated configuration file that identifies one or more remote data processing applications.

37. A distributed data acquisition system comprising:

an interface configured to receive a data acquisition request from a requesting entity;

an information server configured to provide access to one or more local data sources;

one or more shadow information servers, each shadow information server configured to provide access to one or more remote data sources; and

an application server, in communication with the information server and the shadow information server, and configured to fulfill the received data acquisition request by selectively accessing local and remote data sources in a manner that is transparent to the requesting entity.

38. The system of claim 37 wherein the interface configured to receive a data acquisition request from a requesting entity comprises a web server.

39. The system of claim 37 wherein each shadow information server has a communications link for communicating with an interface at a remote data processing system.

40. The system of claim 39 wherein the shadow information server communicates with a servlet executing in a web server at the remote data acquisition system.

41. The system of claim 37 wherein each shadow information server has an associated configuration file that identifies one or more remote data source.

42. A distributed data acquisition and processing system comprising:

an interface configured to receive an information request from a requesting entity;

a processing server configured to provide access to one or more local data processing applications;

one or more shadow processing servers, each shadow processing server configured to provide access to one or more remote data processing applications;

an information server configured to provide access to one or more local data sources;

one or more shadow information servers, each shadow information server configured to provide access to one or more remote data sources; and

an application server, in communication with the processing server, the shadow processing server, the information server, and the shadow information server, and configured to fulfill the received information request by selectively accessing local and remote data sources and local and remote data processing applications in a manner that is transparent to the requesting entity.

43. A method for managing heterogeneous data sources, the method comprising:

a) querying a plurality of heterogeneous data sources, each data source having an associated software wrapper configured to (i) create an object representation of the data, (ii) transform a language of the query into a native language of the data source, (iii) construct a database for caching information contained in the data source, (iv) cache the information contained in the data source in the database automatically; (v) perform self-tests to ensure the wrapper is operating correctly, (vi) provide notification upon detecting an error, and (vii) download and install updates automatically when an error is detected;

b) creating an object representation of each queried data source;

c) normalizing data in the object representations to provide a semantically consistent view of the data in the queried data sources; and

d) aggregating the object representations into a universal data representation.

44. The method of claim 43 wherein normalizing data comprises performing data normalization or vocabulary normalization or both.

45. The method of claim 43 further comprising removing duplicate data.

46. The method of claim 43 further comprising verifying an update's authenticity prior to installation.

47. The method of claim 43 wherein querying the plurality of data sources comprises submitting a query to a data integration engine that distributes the query to the plurality of data sources.